Fuzzy Rule Based Profiling Approach For Enterprise Information Seeking and Retrieval

Obada Alhabashneh
Mutah University
Mutah,Jordan
o.alhabashneh@mutah.edu.jo

Rahat Iqbal
Coventry University
Coventry, UK
r.iqbal@coventry.ac.uk

Faiyaz Doctor
Coventry University
Coventry, UK
faiyaz.doctor@coventry.ac.uk

Anne James
Coventry University, UK
a.james@coventry.ac.uk

Abstract—With the exponential growth of information available on the Internet and various organisational intranets there is a need for profile based information seeking and retrieval (IS&R) systems. These systems should be able to support users with their context-aware information needs. This paper presents a new approach for enterprise IS&R systems using fuzzy logic to develop task, user and document profiles to model user information seeking behaviour. Relevance feedback was captured from real users engaged in IS&R tasks. The feedback was used to develop a linear regression model for predicting document relevancy based on implicit relevance indicators. Fuzzy relevance profiles were created using Term Frequency and Inverse Document Frequency (TF/IDF) analysis for the successful user queries. Fuzzy rule based summarisation was used to integrate the three profiles into a unified index reflecting the semantic weight of the query terms related to the task, user and document. The unified index was used to select the most relevant documents and experts related to the query topic. The overall performance of the system was evaluated based on standard precision and recall metrics which show significant improvements in retrieving relevant documents in response to user queries.

I. INTRODUCTION

The amount of digital information available on the Internet and various Intranets often causes information-overload, significantly increasing the amount of time and cognitive resources needed to acquire relevant and accurate information. Current enterprise search systems produce results mainly based on specific keywords without using the search context effectively [37][42]. These systems do not provide context relevant information to meet the dynamic information needs of enterprise users [67]. Previous research performed by the International Data Corporation on information workers has found that more than 26% of their search sessions failed to produce relevant search results [35]. Moreover, it was estimated that the information workers spend approximately 9% of their time searching for information that did not produce any results. This can lead to lower quality products as well as decisions based on inaccurate and out dated information [14][9] [81] [55] [68] [62].

In order to improve the quality of search results it is crucial to investigate human information seeking behaviour [78][34] which relevance feedback can be used to achieve. The feedback is based on the knowledge of how relevant the particular piece of information is to the user and how its content can be reused in order to find and rank documents that are similar. In general, there exist two techniques of relevance feedback: explicit and implicit [50].

In explicit feedback, users mark the documents as relevant or not relevant whereas in implicit feedback, the relevance is estimated based a behavioural observation such as reading time, click count, etc. A significant number of online companies already take advantage of relevance feedback to provide their customers with relevant domain specific information [49]. For example, Amazon.com and many other e-commerce companies use collaborative filtering as an implicit feedback technique, and use it to recommend other relevant products to the consumer when particular items are purchased. Online music providers such as Last FM creates rich user profiles using their access to the particular sound track listened to by the user. Similarly, Google Mail takes advantage of factors such as response time and emailing frequency to assign priority to emails [49].

Relevance feedback can be used to develop user profiling to enhance the search result [23] [45]. User profiling can be based on various parameters relating to search, tasks or other user contexts [46] [3]. A variety of machine learning approaches can be used to model user profiles based on information seeking needs. These user models should learn and adapt according to user behaviour over time [14]. Fuzzy logic can be used to develop user profiling while handling the uncertainty and ambiguity in user data and fuzzy systems can help to enhance the classification of user relevancy. More precisely, fuzzy sets provide an expressive method for user judgment modelling and fuzzy rules provide an interpretable method for classifying the relevance of information to the user [83][48][32].

In this paper we present a new approach for enterprise IS&R systems which uses fuzzy logic to develop task, user and document profiles to model user information seeking behaviour. Relevance feedback was captured from real users engaged in IS&R tasks and was used to develop a linear regression model for predicting document relevancy based on implicit relevance indicators. Based on the model, fuzzy relevance profiles were created using Term Frequency and Inverse Document Frequency (TF/IDF). Fuzzy rule based summarisation was

used to integrate the three profiles into a unified index reflecting the semantic weight of the query terms related to the task, user and document. The fuzzy rule summarisation technique provides a method for rule extraction and compression [83]. Weighted fuzzy rules were extracted based on the rule quality measures: generality and reliability. The rules provide flexible modelling, which can be adaptable and extendable as more data is accumulated on user search tasks. The generated unified index was used to select the most relevant documents and experts related to the query topic.

The rest of the paper is organised as follows: Section II presents the related literature review, focusing on relevance feedback for enterprise IS&R and fuzzy logic approaches for user profiling. Section III describes the proposed approach using task, user and document profiles. Section IV discusses experiments and results. Section V presents the evaluation of the proposed approach. Three types of evaluation are discussed: method validation; precision and recall; and comparative analysis. Finally conclusions and future directions are presented in Section VI.

II. LITERATURE REVIEW

A. Relevance feedback

Individual and/or group profiles are used by intelligent search systems in order to produce better search results based on the level of interest of the user in the search topic. Relevance feedback is the main data source for constructing these user profiles. Relevance feedback has been investigated by several researchers [69] [9] [61] [53]. Previous research has analysed user behaviour and found a significant relationship between the time spent on reading Usenet news and the interest level of the user. This was proven by comparing observational studies with explicit interest measures [38] [51] [61] [53].

Current research shows that the combination of several relevance feedback parameters can produce better results [38], [82], [16], [10] and [24]. It was found that reading time, along with other user behaviour can be a very reliable indicator of content relevancy. It was noticed that even though there is a positive correlation between mouse movement and amount of clicks, reading time was also shown to be a reliable indicator of user interest [38]. The experiments showed that the integration of multiple implicit parameters such as dwell time, click-through, text selection and page review can produce better results in predicting document relevancy [82].

In another research study, the relationship between user behaviour during the dwell time on the search engine results page (SERP) and the relevancy of the page was investigated [16]. The experiments showed that cursor movements and scrolling were more effective than considering the dwell time alone to estimate the page (document) relevancy. Similarly, search performance was enhanced significantly by using text-selection data [10]. User post-click behaviour parameters such as mouse clicks, mouse movements, text selection and cursor trails were also used to cluster the users based on their behaviour similarity [24].

A document can be represented by using the vocabulary used by the user during the retrieval of the document [66]. The content-based (TF-IDF) and the connectivity-based (PageRank) ranking algorithms using the click-through data were integrated to improve the search result for a web page by one researcher [14]. Another approach proposed the development of a snippet-based algorithm to estimate the document relevance which was found to be more efficient than the approach used for commercial search engines [57]. Another post-click parameter which has been deemed to be useful is page review or re-finding. It is argued that about 30% of the user queries retrieve a page which the user has previously visited [76]. A page review based algorithm was proposed to predict the page relevancy which was shown to significantly improve the retrieval performance [77].

A.1. Relevance Feedback for Enterprise IS&R

A number of approaches enhance the retrieval performance of enterprise IS&R systems based on relevance feedback, [31] [72] [28][13] [80] [56]. The user annotation based approach was suggested to enhance the retrieval performance using the PageRank algorithm which was commonly used in the web search [31]. This approach was shown to slightly improve the retrieval performance when used for enterprise web documents. However, the approach was not applied to non-web documents which are the main corpus for enterprise documents.

A semantic approach was proposed for the search in small and micro size enterprises to extract hidden knowledge in emails and content management systems using tags and annotations provided by the user [72]. The captured tags and annotations were used to build a lightweight semantic web to represent the relationship between documents required for different tasks. The approach showed a good retrieval performance on a small data set.

A knowledge cloud concept was introduced to extract the keywords from enterprise information sources such as documents from content management systems, emails and database applications [28]. These keywords were used as candidate tags for the relevant information. The candidate tags were filtered and ranked based on the taxonomy provided by the organisation, together with Wikipedia topic headings, in order to enhance the search and rank the documents. This research lacked evaluation results.

An automated semantic query-rewrite rule suggestion system was developed to help enterprise IS&R users to write better queries [13]. In the proposed system, the suggestions were created based on a set of rules which were extracted from the co-occurrence of the terms in the query history of successful queries. The proposed approach was shown to improve the retrieval performance and the user satisfaction.

In another research study, an entity-centric query expansion approach was proposed to address the information overloading problem in the enterprise [56]. This approach expanded the user query based on the relevant entities. The entities were previously identified and extracted from enterprise documents using an organisational dictionary, tags which were previously extracted from enterprise web pages, and user annotations. The similarity between the user query and each extracted entity was calculated and then the relevant entities were used to expand the user query. The proposed approach was shown to improve the retrieval performance of the enterprise IS&R.

A class based personalised approach for the enterprise IS&R was suggested to classify documents based on the taxonomy of the organisation, and each document was assigned a particular class [80]. During the search process the users were asked to rate the documents returned by the search according to their relevance to the user query. Based on the document class and the user rating the relevance between

the user and the document class was calculated. The user was then modelled by assigning a number of classes, which were used to filter the search result in the next query. The experimental results showed that the class based user model accurately represented the user interest.

Expert search, one of the main tasks in enterprise IS&R, is attracting an increasing amount of attention from the research community. It has been studied by a number of researchers in different contexts including the enterprise corpora [35], sparse data university environments [11], online knowledge communities [79] and digital libraries [36]. People search can be categorised into profile-based and document-based approaches [12]. In the profile-based approach a profile is created for each user based on the documents they visit, create or author and the user is given a rank based on matching between the profile and the given user query. In the document based approaches the search begins by finding the relevant documents and then retrieving the names of the experts who have knowledge of the information contained in these documents. In general, profile-based methods have a lower component cost than document-based methods as they use a smaller size virtual document to model the user rather than the content of the actual document [75]. On the other hand, the document-based methods aremore effective in ranking people with knowledge of individual documents and require less data management than the profile-based methods [75].

The development of the profile-based approach for expert search has been discussed in [11]. Term-based profiles were created for the candidate users to model their expertise and then were used to retrieve and rank the users based on the relevance of their profiles to the user query. In the same paper [11], a document-based approach for expert search was proposed in which a language model was employed to find the people who had knowledge of the query topic. The document-based model ranked people based on the relevance to the given query of both their profiles and the relevant documents. The relevance between the people and the documents was calculated based on the terms co-occurrence and the order of the co-occurred terms, and the experimental results showed the document-based approach outperformed the profile-based approach.

A probability approach has been proposed to rank relevant people to a user query [39]. The approach combined the traditional relevance model (which calculated the relevance of the document to the query term based on the term frequency of the document), together with the co-occurrence model (which considered the co-occurrence of the query terms in the document in calculating the relevance of the document).

Another interesting research output integrated the information retrieval and graph based approaches to form a hybrid approach for ranking expertise [41] [29]. This approach combined information from social media, online communities and forums with the document-based model to rank the expertise of people for a specific topic. The integration of these two approaches improved the retrieval performance beyond what was possible with each of the individual approaches. In another research study a voting technique was used to improve the people search [59]. The voting techniques were borrowed from the data fusion field and were applied to enhance the retrieval performance of the experts. The proposed voting-based approach was shown to improve the retrieval performance of the people search. A multi-view fuzzy ontology information retrieval model was proposed to handle queries in different domains which involve a high level of subjectivity and uncertainty. The model ranked the retrieved documents based on their relevance degree, confidence degree, and updating degree [8]. The implementation of an efficient fuzzy based information system was exemplified in [73] where the developed system used fuzzy logic to calculate the similarity between the indexed documents and the user query.An enterprise recommender system, called Meven, was used to recommend experts by using the content from the enterprise social web to create a trust matrix between colleagues and to group them according to whether they had similar interests and behaviour. [1].

Some researchers have used PageRank [63] for expert search . In [85] PageRank was used to develop a coupled random walk approach in which citation networks were combined to rank authors and documents. PageRank has also been used to calculate the authority and contribution of experts to a specific topic in online communities [79]. Similarly, in another study, PageRank was used to rank comments and posts from the chains of friends in social networks and online communities [30] in order to estimate the level of knowledge people had of a particular topic.

B. Fuzzy Profiling for Information Retrieval and Filtering.

User profiles have become an important component of intelligent information access systems for information retrieval and filtering in recommender systems [54]. Fuzzy logic has been widely used for developing user profiles to provide a more representative method for user modelling that can handle uncertainty and ambiguity in the relevance feedback. This section presents a number of fuzzy based profiling approaches which could be used for both information retrieval and filtering.

In [21] a fuzzy based user profiling approach was proposed to enhance user clustering in Web data usage . The fuzzy sets were used to approximate the similarity between the preferences of users. In another research study, fuzzy logic was used to infer the degree of genre presence in a movie using tags created by different users to describe the movies [6]. A fuzzy weighting approach was proposed to provide a learning mechanism for user preferences in memory-based recommendation systems [4].

A fuzzy recommendation method called ‘single individual’ was proposed to create recommendations recursively based on the profile of the user in [84]. The fuzzy sets were used to model the recommended object as well as justifying the recommendations based on the similarity of the user preferences. In [26] a fuzzy based conceptual framework for recommending one-and-only items was proposed . One-and-only items were the items which had only one occurrence in the data. The single occurrence of such items placed limits on the ability of classic collaborative filtering to recommend the required item. Fuzzy logic was used for modelling the user preferences for the similarity calculation and a collaborative filtering based algorithm was proposed in which the linguistic labels and the associated fuzzy sets were used to handle the uncertainty and inaccuracy in ranking and recommending items [20]. Fuzzy logic has been used in a hybrid recommendation system to model the interest of an individual in a specific item (e.g. movie) in order to recommend that item to other individuals who have similar interests[70] [65] [18]. A fuzzy based ontology approach was used to represent the degree of trust between users in [60]. The approach used multi-granular fuzzy linguistic modelling to increase flexibility when representing different concepts with different linguistic labels.

Researchers have proposed a fuzzy based agent to rank and recommend candidates' CVs within recruitment systems [32]. Fuzzy logic was used to model the job preferences of the selection board members and also to resolve the uncertainty and conflict in the group decision making. Granular fuzzy sets have been used to provide more flexible means of preferences modelling [64]. For instance, they were used to model experts' preferences for group decision making [80]. A granular fuzzy based approach was also proposed to support consensus in group decision making in [17]. The granular fuzzy format was used to model the preference relations to represent the opinions of the decision makers. Being more abstract, it allowed the required level of flexibility to achieve agreement.

In another research study, a fuzzy based method that improved the collaborative filtering efficiency in the context of multiple collaborating users was proposed [33]. In this system, the fuzzy sets were used to normalise the values of the user preferences in order to calculate the similarity between the users. The preferences of the user might have different data types and ranges, so the membership function of the fuzzy sets was used to normalise the preference values to a number between 0 and 1. In addition, fuzzy logic was used for ranking and scoring items. In [39], a fuzzy based ranking mechanism was proposed. The ranking function was based on the computation of term frequency, inverse document frequency and the normalization of the user query terms in the documents. In more recent research [8], a fuzzy based ranking function was proposed to provide more efficient information retrieval. The function used fuzzy logic to compute the term weight based on many different weighting schema including: term frequency, inverse document frequency and normalization. A combined fuzzy based similarity measure was proposed to overcome the limitation of the conventional measures such as Cosine, Jaccard, Euclidean and Okapi-BM25 [40]. In the approach, a fuzzy rule base was used to infer a unified similarity value from the conventional measures. A fuzzy based modified information retrieval system was proposed [73] to extend the function of the information retrieval system for forecasting the future trading values of the stock market.

The existing approaches described above were mainly focused on identifying indicators of document relevancy and user preference. They do not consider combining these with user profiling, nor in particular the relevance of information to a particular task which is required in an enterprise IS&R system. Also, many of these approaches focussed on well described contents such as news stories, events, and movies and not on the unstructured contents which are commonly found in enterprise systems. Such enterprise contents have less descriptive detail. In addition, with these systems, the perception of the relevance of retrieved data retrieved by the user can be subjective and inconsistent.

III. PROPOSED APPROACH

Our proposed approach is based on relevance feedback and fuzzy logic. Relevance feedback is used as the main data source for developing a task, user and document profile from the user query. The user profile models the user interest, the document profile contains the terms which were used by different users to retrieve the document and the task profile contains the terms which were used by different users to complete a work related task. Every task, user and document is modelled as a set of weighted terms in their associated profiles. These are extracted from the terms used in user queries with the term weight reflecting the relevance level of that term to the user, document and task profile. Relevance feedback is usually associated with a *document visit* when the user gives their feedback on the relevance of the retrieved document to the information required. More precisely, the document visit is not only associated with the user, but also with the work task for which the document was visited as well as the visited document itself. User, task and document profiles are then combined to produce a unified term-visit instance index containing a unified and normalised term weight for the associated documents.

Since relevance feedback involves a high level of uncertainty due to the inconsistency of user behaviour and the subjectivity in their assessment of relevancy [37], handling such uncertainty is crucial for achieving better performance. A fuzzy approach is used to overcome the uncertainty and bias in user judgment. This approach provides a normalized ranking method for recommendations in enterprise IS&R and also enables the adaptability of the system, through which the system becomes sensitive to the changes in the user behaviour or interest. Our approach consists of six phases as shown in Fig. 1. *Phase 1: Relevance Feedback Collection. Phase 2: Document Relevance Prediction. Phase 3: Fuzzy Based Task, User and Document Profiling*. *Phase 4: Fuzzy combined weight calculation. Phase 5: Recommendation of documents and experts. Phase 6: Recommendation Presentation*

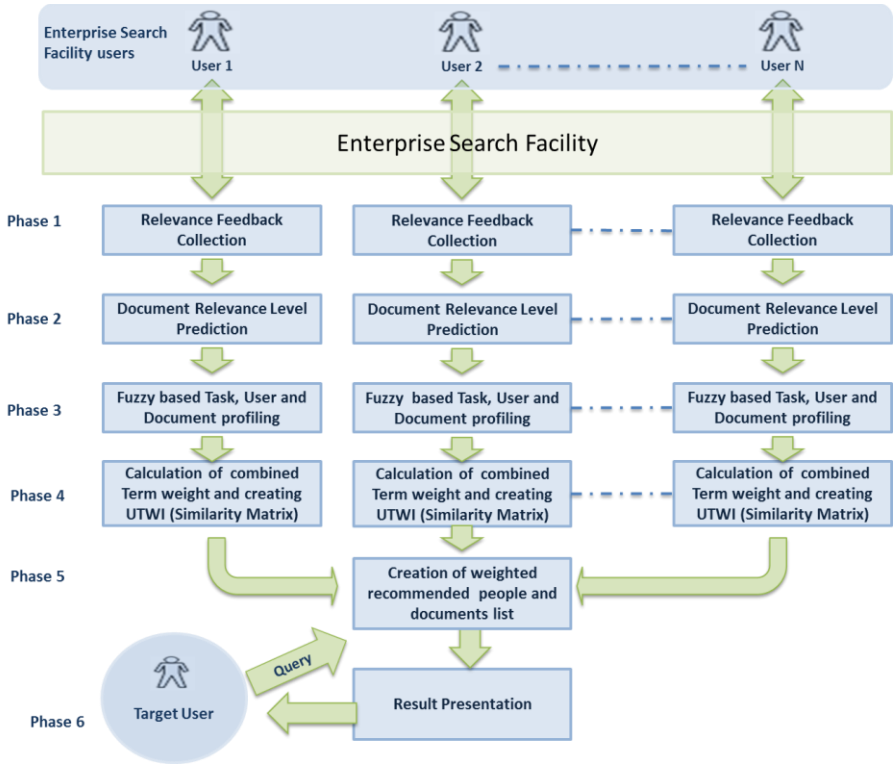


Fig. 1: The Proposed Approach

Phase 1: Relevance Feedback Collection

In this phase the relevance feedback is captured from the users during the search process through a plug-in that works as an upper layer on top of the search facility being used in the enterprise as shown in Fig. 2. The captured relevance feedback includes implicit parameters, explicit parameters, user queries and interaction features. The implicit parameters include: visit time stamp, time on page, number of mouse clicks, mouse movement, mouse scrolling, scroll bar holding, key down times, key up times, bookmark, save and print. Explicitly, the users are asked to rate the visited documents indicating their relevance to the query/task. The users and their tasks are identified through unique user IDs. The query information includes: query text, query time stamp, and number of documents retrieved by the query. Interaction features include document ID and document hyperlink for each visited document.

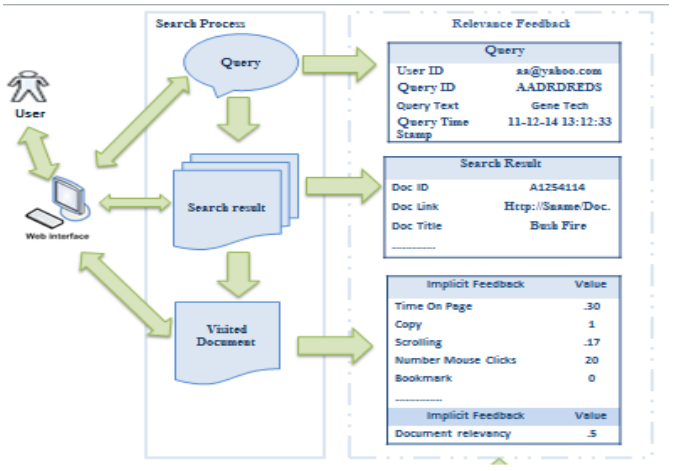


Fig. 2: Relevance Feedback Collection

Phase 2: Document Relevance Prediction

In this phase the relevance level of the visited documents is predicted from the implicit feedback parameters. The predicted value was calculated, as shown in TABLE I, using a linear predictive model based on linear regression analysis. The accuracy of the model was validated using the R-squared ($\boldsymbol{R^2}$) measure [27] and it achieved an accuracy of 76.5 % . The model was developed using the following steps:

Step1, The implicit and explicit parameters are categorised into independent variables (IV) or ***Predictors*** and dependent variable (DV) or ***Target*** as shown in Fig. 3. Predictors' values are used to estimate the value of the target.

Step2, The correlation analysis is carried out to identify which implicit relevance feedback parameters are correlated with the explicit document relevancy. Correlation analysis is carried out in order to identify implicit parameters which have a significant relationship with explicit user feedback. Identifying these parameters decreases the dimensionality of the data by excluding the parameters that have no correlation with the explicit relevance level from the regression analysis in the next step. IBM-SPSS-Statistics Version 22 was used to automatically carry out the correlation analysis. While the data was imported manually into IBM-SPSS-Statistics for experimental purposes, it could be carried out automatically by the application program interface (API's). TABLE I shows the results of the correlation analysis.

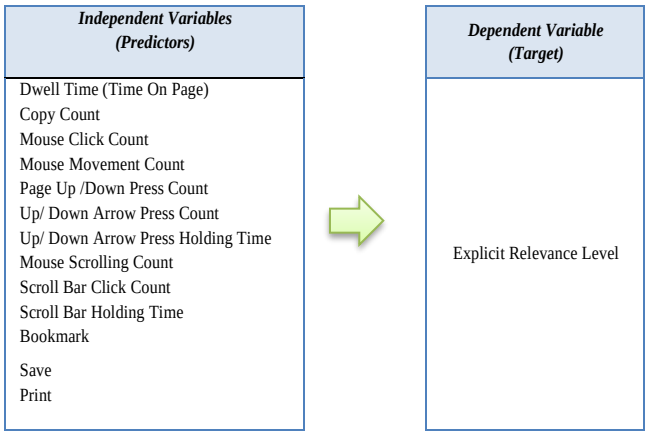


Fig. 3: Predictors and Target

TABLE I : CORRELATION ANALYSIS OF PREDICTORS AND TARGET

Implicit Feedback Parameters (Predictors)	**Explicit Relevance Level (Target)**
Time On Page	.144**
Copy Count	-.030
Mouse Click Count	.170**
Mouse Movement Count	.202**
Page Up Down Press Count	-.007
Up Down Arrow Press Count	-.032
Up Down Arrow Press Holding Time	-.027
Mouse Scrolling Count	.189**
Scroll Bar Click Count	-.042
Scroll Bar Holding Time	-.038
Bookmark	.283**
Save	.143**
Print	.014
Visit Time stamp	.083**
Explicit Relevance Level	1
**. Correlation is significant at the 0.01 level (2-tailed). * Correlation is significant at the 0.05 level (2-tailed).	

As shown in the TABLE I, the candidate implicit parameters for a linear relationship with the explicit relevance level of the document include: dwell time (time on page), mouse click count, mouse movement count, muse scrolling count, save, print and visit time stamp. These parameters are considered in the regression analysis in Step 3.

Step3, The relevance level of the visited document is predicted from the implicit feedback parameters. The predicted value is calcula ted by the linear predictive model, which is developed using linear regression analysis. In linear regression, regression models involve three types of parameters; *Coefficients (β)* which are the unknown parameters, *Predictors (X)* which are the independent variables and *Target (Y)* which is the dependent variable. A linear regression model relates Y to a function of X and β [27]. Equation (1) shows this relationship.

$$\boldsymbol{Y} \approx f(\boldsymbol{X}, \boldsymbol{\beta}) \tag{1}$$

The approximation is usually formalized as $\boldsymbol{E(Y \mid X) = f(X, \beta)}$. In general, a multiple linear regression model with $\boldsymbol{N}$ independent variables and one dependent variable is defined as shown in Equation (2).

$$\hat{Y} = \beta_{\circ} + \sum_{i=1}^{N} \beta_i X_i \tag{2}$$

Where $\hat{Y}$ is the fitted predicted value of the dependent variable, $\beta_{\circ}$ is the intercept, β_i is the variable coefficient, X_i is the value of an independent variable, N is number of the independent variables.

The candidate implicit parameters which were identified in Step 2, were used as predictors in the regression analysis to discover a linear relationship between any of these parameters and the explicit relevance level. Only dwell time, mouse scroll count and mouse movement count were found to have a linear relationship with explicit relevance level. The regression analysis was carried out using IBM-SPSS-Statistics Version 22. TABLE II shows the result of the analysis. Only the variables with a significance level ≥ .05 are considered as predictors in the predictive model.

TABLE II. COEFFICIENTS FOR THE TARGET EXPLICIT RELEVANCE FEEDBACK

Model Term		*Coefficient (β_i)*		*Sig*	*Importance*
Intercept	($\beta_{\circ}$)	1.395		.000	-
Dwell (Time on Page)	(X_1)	0.069	(β_1)	.021	0.893
Mouse Scroll Count	(X_2)	0.013	(β_2)	.012	0.079
Mouse Movement Count	(X_3)	0.113	(β_3)	.031	0.028

Substituting the values from the table into Equation (2), the predictive linear model for explicit relevance level becomes:

$$\hat{Y} = 1.395 + (X_1 \times 0.069) + (X_2 \times 0.069) + (X_3 \times 0.069) \tag{3}$$

Then the importance of each predictor is used to normalize the value:

$$\hat{Y} = 1.395 + (X_1 \times 0.069 \times 0.893) + (X_2 \times 0.069 \times 0.079) + (X_3 \times 0.069 \times 0.028) \tag{4}$$

Predictor importance is an automated value which is calculated by the automatic linear modelling tool in IBM-SPSS-Statistics. This value indicates the relative importance of the input variable in predicting the output variable [43].The tool calculates the predictor importance using the leave-one-out method [7] and based on the residual sum of squares (SSe) [44].

Phase 3: Fuzzy Based Task, User and Document Profiling

In this phase, the task, user and document profiles are created by employing an adaptive fuzzy approach. *Fig. 4* gives an overview of the steps involved in the process of creating the profiles. In this approach, the state of the art method proposed in [55] was adapted to suit the query text analysis. The following sub sections describe the construction of the three profiles.

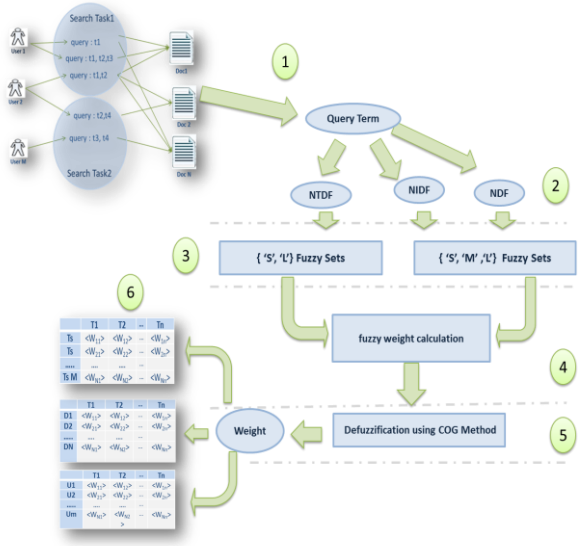


Fig. 4: Fuzzy based Task, User and Document Profiling

1. User Profile

The user profile was developed using the relevance feedback captured during the completion of search tasks. In the profile, each user

is represented as a set of weighted terms which represent their interests. More precisely, the following steps were followed to develop the user profile:

Step 1, The set of queries $\boldsymbol{Q}$ which led to document visits is selected.

Step 2, The set of all users $\boldsymbol{U}$ **is selected** and for each user ($\boldsymbol{U_K} \in \boldsymbol{U}$)**'** steps (3-9) are carried out-.

Step 3, A subset Ω_{U_k} of the query set $\boldsymbol{Q}$ is identified by selecting the queries that the user $\boldsymbol{U_k}$ has created.

Step 4, After identifying the sets Ω_{U_k} in step 3, the queries in the set are pre-processed and transformed into a set of candidate terms through eliminating stop-words and stemming by Porter's algorithm [67].

Step 5, The frequency measures, Distributed Term Frequency (DTF), Document Frequency (DF), and Inverse Document Frequency (IDF), of each candidate term are calculated and normalized based on each set Ω_{U_k} and used as inputs to a fuzzy system for calculating a weight for each term.

These frequency measures are used to calculate the term frequency in a document collection. They are also used in a collection of user queries where each user query could be considered as a document in order to calculate the frequency of the query terms [22][66]. Based on that, in this step only, both terms, '*document*' and '*query*', refer to the user query. The DTF reflects the frequency and distributed status of a term in a set of user queries. This is calculated by dividing total occurrences of the term in the query set Ω_{U_k} by the number of the queries which contain the term in the set Ω_{U_k}. The DF represents the frequency of queries having a specific term within the set $\boldsymbol{Q}$. The Normalized Distributed Term Frequency (NDTF) is defined in Equation (5).

$$NDTFi = \frac{\frac{TF_i}{DF_i}}{Max_j \left[\frac{TF_j}{DF_j}\right]} \tag{5}$$

where, $\boldsymbol{TF_i}$ is the frequency of term $\boldsymbol{t_i}$ in the query set Ω_{U_k}, $\boldsymbol{DF_i}$ is the number of queries having term $\boldsymbol{t_i}$ in the query set Ω_{U_k}, $\boldsymbol{i}$ and $\boldsymbol{j}$ = 1 to *M* where *M* is the number of the terms in the set Ω_{U_k}. NDF is defined in Equation (6).

$$NDFi = \frac{DF_i}{Max_j\, DF_j} \tag{6}$$

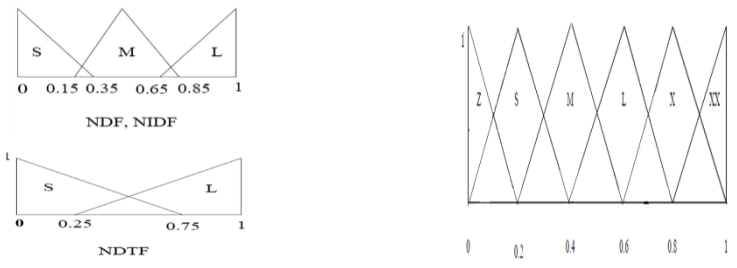
where $\boldsymbol{DF_i}$ is the number of queries having term $\boldsymbol{t_i}$ in the in the query set Ω_{U_k}.

The IDF represents the frequency of the term in the query set $\boldsymbol{Q}$ rather than the set Ω_{U_k}. We used IDF to identify the terms which appear in many queries which might relate to different tasks, users and documents. These terms are not very useful for representing the relevance level and consequently they will be given a less weight than the others. The Normalized Inverse Document Frequency (NIDF) is defined as follows:

$$NIDFi = \frac{IDF_i}{Max_j\, IDF_j}\,, IDFi = Log\, \frac{N}{n_i} \tag{7}$$

where, *N* is the total number of queries in $\boldsymbol{Q}$ and $\boldsymbol{n_i}$ is the number of queries in $\boldsymbol{Q}$ in which the term $\boldsymbol{t_i}$ appears.

Step 6, The crisp values of the three input variables (NDTF, NDF, and NIDF) are fuzzified and mapped onto predefined fuzzy sets. As shown in *Fig. 5. a)*, NDF and NIDF have three linguistic labels { S(Small), M(Middle), L(Large) }, and NTDF has two linguistic labels{ S(Small), L(Large) }. As shown in *Fig. 5. b)*, the output variable *TW* has six fuzzy sets associated with six linguistic labels {Z(Zero), S(Small), M(Middle), L(Large), X(Xlarge), XX(XXlarge) }.



a).Input Variables b). Output Variable *TW*

Fig. 5: Fuzzy Sets for Input Variables

Step 7, The 18 '$\boldsymbol{If \rightarrow Then}$' fuzzy rules which are described in [71] are used to infer a fuzzy term weight (*TW*) for the term $\boldsymbol{t_i}$. These rules are constructed based on the assumption that the important or representative terms may occur across many queries in the representative query set Ω_{U_k} but not in the whole selected query set $\boldsymbol{Q}$. In other words these terms have high NDF and NIDF values and low NDTF

values. For example, as shown in *Fig. 6*, when NDF of a term is high and its NIDF is also high, the term is considered as a representative keyword so the output weight is between X and XX.

NDF \ NIDF	S	M	L
S	Z	Z	S
M	Z	M	L
L	S	L	X

NDTF = S

NDF \ NIDF	S	M	L
S	Z	S	M
M	Z	L	X
L	S	X	XX

NDTF = L

Fig. 6: *WT* Calculation Fuzzy Rules

Step 8, The output of step 7, *TW*, is de-fuzzified using the center of gravity (COG) method in order to get a crisp weight $TW_{U_k t_i}$ for each term to be added to the profile associated with the collection Ω_{U_k}.

Step 9, The term t_i with its weight $TW_{U_k t_i}$ are added to the profile being created. However, as the system is used, more relevance feedback, including user queries will be captured and the system will calculate new weights if the term frequencies change. The profile will then be updated by changing the term weight(s) to the new value(s). More formally, if P_{U_k} is the profile associated with the collection Ω_{U_k} and M is the number of terms in Ω_{U_k}, then the profile P_{U_k} is defined as a set of weighted terms as shown in Equation (8).

$$P_{U_k} = \bigcup_{i=1}^{M} t_i TW_{U_k t_i} \tag{8}$$

As described in the previous steps, this phase creates /updates user profiles. The profiles are stored in a database table as shown in the TABLE III.

TABLE III : SAMPLE OF USER PROFILE

User ID	*Term*	*Term Weight*
***569@coventry.ac.uk	OCEAN	0.571067496
***569@coventry.ac.uk	CURRENT	0.532861611
***569@coventry.ac.uk	CONDITION	0.50125059
***569@coventry.ac.uk	DIET	0.592570996
***569@coventry.ac.uk	ASSESSMENT	0.57609926
***569@coventry.ac.uk	PRODUCT	0.564558761
***569@coventry.ac.uk	LIFECYCLE	0.564558761
***569@coventry.ac.uk	SENSOR	0.552254561

2. Task Profile

In the task profile, each task is represented as a set of weighted terms that have been used to complete the search task. The weight reflects the relevance between each of these terms and the search tasks. The following steps are followed to develop task profile.

Step 1, The set of queries $\boldsymbol{Q}$ which led to document visits is selected.

Step 2, The set of all Tasks $\boldsymbol{S}$ anis selected and for each task ($\boldsymbol{S_y} \in \boldsymbol{S}$ **),**

steps (3-9) are carried out.

Step 3, A subset Ω_{S_y} of the query set $\boldsymbol{Q}$ is identified by selecting the user queries that were created to complete task $\boldsymbol{S_y}$.

Step 4, After identifying the sets Ω_{S_y} in step 3, the queries in the set are pre-processed and transformed into a set of candidate terms through the same method as mentioned in step 4 of user profile.

Step 5, The term frequency measures, Distributed Term Frequency (DTF), Document Frequency (DF) and Inverse Document Frequency (IDF), of each candidate term were calculated and normalized based on each set Ω_{S_y} and used as inputs to a fuzzy system for calculating a weight for each term.

Step 6, The crisp values of the three input variables (NDTF, NDF, and NIDF) are fuzzified and mapped in the same way as in step 5 of user profile subsection.

Step 7, The 18 '***If →Then***' fuzzy rules which are described in step 7 of the user profile construction method are used to infer a fuzzy term weight (*TW*) for each term t_i.

Step 8, The output of step 7, *TW*, is de-fuzzified using the center of gravity (COG) method in order get a crisp weight $TW_{S_y t_i}$ for each term to be added to the profile associated with the collection Ω_{S_y}.

Step 9, The term t_i with its weight $TW_{S_y t_i}$ is added to the profile being created. However, as the system is used, more relevance feedback, including user queries will be captured and the system will calculate new weights if the term frequencies change. The profile will then be updated by changing the term weight(s) to the new value(s).

More formally, if P_{S_y} is the profile associated with the collection Ω_{S_y} and M is the number of terms in Ω_{S_y} then the profile P_{S_y} is defined as a set of weighted terms as follows:

$$P_{S_y} = \bigcup_{i=1}^{M} t_i TW_{S_y t_i} \quad (9)$$

As described in the previous steps, this phase creates / updates the tasks profile. The profiles were stored in a database table as shown in the TABLE IV.

TABLE III: SAMPLE TASK PROFILE

Task Id	*Term*	*Term Weight*
T1	TECHNOLOGY	0.363189988
T1	GENE	0.309313589
T1	RNAI	0.129074074
T1	MODIFY	0.094116972
T1	COTTON	0.090064157
T1	BIOTECHNOLOGY	0.088960647
T1	GENETICALLY	0.077359248
T1	FOCUS	0.039090293

3. Document Profile

In the document profile, each document is represented as a set of weighted terms that have been used by the users to retrieve the document relevant to their tasks. The weight reflects the relevance between each of these terms and the documents. The following steps were followed to construct document profile.

Step 1, The set of queries $\boldsymbol{Q}$ which led to document visits is selected.
Step 2 The set of all visited Documents $\boldsymbol{D}$ is selected and each for each task $\boldsymbol{D_g} \in \boldsymbol{D}$ steps (3-9) are carried out.
Step 3, A subset $\Omega_{\boldsymbol{D_g}}$ of the query set $\boldsymbol{Q}$ is identified by selecting the user queries that have led to visit the document $\boldsymbol{D_g}$.
Step 4, After identifying the set $\Omega_{\boldsymbol{D_g}}$ in step 3, the queries in the set were pre-processed and transformed into a set of candidate terms through the same method mentioned in step 4 of user profile.
Step 5, The term frequency measures, Distributed Term Frequency (DTF), Document Frequency (DF) and Inverse Document Frequency (IDF),of each candidate term are calculated and normalized based on each set $\Omega_{\boldsymbol{D_g}}$ and used as inputs to a fuzzy system for calculating a weight for each term.
Step 6, The crisp values of the three input variables (NDTF, NDF, and NIDF) are fuzzified and mapped in the same way in step 5 of the user profile construction method.
Step 7, The 18 '$\boldsymbol{If \rightarrow Then}$' fuzzy rules which are described step 7 of the user profile construction method are used to infer a fuzzy term weight (*TW*) for the term $\boldsymbol{t_i}$.
Step 8, The output of step 7, *TW*, is defuzzified using the center of gravity (COG) method in order get a crisp weight $TW_{D_g t_i}$ for each term to be added to the profile associated with the collection $\Omega_{\boldsymbol{D_g}}$.
Step 9, The term t_i with its weight $TW_{D_g t_i}$ is added to the profile being created. However, as the system is used, more relevance feedback, including user queries will be captured and the system will calculate new weights if the term frequencies change. The profile will then be updated by changing the term weight(s) to the new value(s). More formally, let's assume P_{D_g} is the profile associated with the collection Ω_{D_g} and M is the number of terms in Ω_{D_g} then the profile P_{D_g} is defined as a set of weighted terms as follows:

$$P_{D_g} = \bigcup_{i=1}^{M} t_i TW_{D_g t_i} \quad (10)$$

As described in the previous steps, this phase creates/updates document profiles. The profiles were stored in a database table as shown in the TABLE V.

TABLE V: SAMPLE OF THE DOCUMENT PROFILE

Doc URL	*Term*	*Term Weight*
CSIRO000/CSIRO000-00000000.html	DIET	0.592570996
CSIRO000/CSIRO000-00000000.html	WELLBEING	0.520925064
CSIRO000/CSIRO000-00000000.html	TOTAL	0.511938243
CSIRO000/CSIRO000-00000000.html	DEVELOPMENT	0.5
CSIRO000/CSIRO000-00000000.html	DIETARY	0.5
CSIRO000/CSIRO000-00000000.html	TRIAL	0.5
CSIRO000/CSIRO000-14537203.html	HUMAN	0.592559456
CSIRO000/CSIRO000-14537203.html	CLINIC	0.53027557

Phase 4: Fuzzy Combined Weight Calculation

In this phase, the task, the user and the document profiles are combined in one index which is called the Unified Term Weight Index

(UTWI). In this index each term has a unified weight per task, per user and per document. If term t_i was used by user U_k to retrieve the document D_g in order complete the search task Sy then the unified term weight for t_i is W_{iykg}. This means that the new weight considers the relevance between the term and the whole combination of the three factors; the user, the document and the task. This phase includes the following steps:

Step1, *Fuzzy rules Extraction*, if *V* is the set of document visits in the data set which contains *H* visits, then V_h is the document visit instance where h=1 to *H*. Each V_h is associated with the user query Q_e which led to this visit where e=1 to E and E is the number of queries in the dataset, the search task in which it occurred S_y, the user who made this visit U_k, the visited document D_g and the predicted relevance feedback R_h(see output of phase 2). Q_e consists of Z terms where t_{ez} is the query term in Q_e and z =1 to Z. Each t_{ez} is associated with its weight W in each of the profiles of S_y, U_k, and D_g that were computed in phase 3. These three weights are associated with the predicted relevance R_h. As a result, each term t_{ez} is represented as a set of four values $\{W_{s_y t_{ez}}, W_{u_k t_{ez}}, W_{d_g t_{ez}}, R_h\}$. If we consider the three first weights as inputs and the R_h as a result, then we have a sequence of three input values and one result value $\{W_{s_y t_{ez}}, W_{u_k t_{ez}}, W_{d_g t_{ez}} \rightarrow R_h\}$ for each instance in dataset.

The inputs and output values are mapped to predefined fuzzy sets with the linguistic labels 'Low' (L), 'Medium' (M) and 'High' (H) based on Mendel Wang method described in [25] as shown in *Fig 7*. In our system, the shapes of the membership functions (MFs) for each fuzzy set are based on triangle MFs as shown in *Fig. 7*. A triangular MF is specified by three parameters {a, b, c} as in Equation (11). *Fig.7* illustrates triangular MFs defined for the fuzzy sets.

$$Triangle(x; a; b; c) = \begin{cases} 0, & x \le a. \\ \dfrac{x-a}{b-a}, & a \le x \le b. \\ \dfrac{c-x}{c-b}, & b \le x \le c. \\ 0, & c \le x. \end{cases} \tag{11}$$

where the parameters $\{a, b, c\}$ (with $a<b<c$) determine the x coordinates of the three corners of the underlying triangular MF.

The outcome from this step is a set of antecedents and consequents also called '***if →then'*** fuzzy rules where each of the inputs and the outputs are represented by the associated linguistic label as shown in TABLE VI. If B is the linguistic label {'L','M','H'} of the value of each of the inputs and the output then the fuzzy rule FR_h is:

$$B\left(W_{s_y t_{ez}}\right), B\left(W_{u_k t_{ez}}\right), B\left(W_{d_g t_{ez}}\right) \rightarrow B(R_h) \tag{12}$$

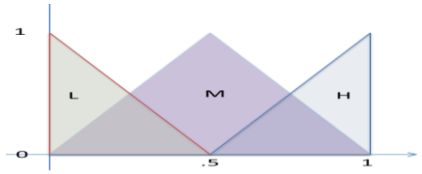


Fig. 7: Fuzzy Sets for Input and Output Variables

Step 2, *Compression of Fuzzy Rules*, a rule compression on the fuzzy rules resulting from the previous step is performed in order to extract those rules with the maximum firing strength. This process involves a modified calculation of two rule quality measures from which the scaled fuzzy weight is derived for each unique summarisation rule. The quality measures are based on generality which measures how many data instances support each rule and reliability that measures the confidence level in the data supporting each rule [48].

In our approach the rule generality is measured using scaled fuzzy support and the reliability of the rule is based on its scaled confidence level. The fuzzy support of a rule is calculated as the product of the rule's support and firing strength. The support of a rule refers to coverage of data patterns that map to it, while its firing strength measures the degree to which the rule matches those input patterns.

TABLE IVI: SAMPLE OF THE EXTRACTED FUZZY RULES

User	Task	Document	Term	W_u	W_T	W_D	→	R
U1	T2	11884897.html	PUBLICATION	H	M	M	→	L
U2	T20	09530858.html	PUBLICATION	M	M	M	→	H
U3	T3	15292585.html	DIFFER	M	M	M	→	M
U4	T3	15292585.html	SHEET	M	M	H	→	M
U5	T3	15292585.html	TRIAL	M	M	M	→	M
U3	T6	01314419.html	DIETARY	M	M	H	→	L
U5	T2	01314419.html	TOTAL	L	M	M	→	H
U6	T17	03452997.html	LEAD	M	M	H	→	H
U7	T10	11659583.html	TOTAL	M	M	H	→	L
U8	T6	03288103.html	CARBON	L	M	H	→	M
U9	T2	13286918.html	FIRE	H	M	H	→	L
U2	T20	01037988.html	COPPER	M	M	H	→	M
U10	T17	04945868.html	CARBON	L	M	H	→	H
U10	T17	04945868.html	PROJECT	L	M	M	→	H
U11	T11	04945868.html	YARN	L	M	M	→	L
U10	T17	04945868.html	YARN	M	M	M	→	H
U6	T19	13878470.html	BITE	M	M	M	→	L

The fuzzy support for the rule can be used to identify the unique rules together with the most frequent occurrences of data patterns associated with them, where the data patterns also most closely map to those rules. The fuzzy support for each rule is scaled based on the total data patterns for each output (consequent) so that the frequencies are scaled in proportion to the number of data patterns found in each consequent set. The calculation of the scaled fuzzy support for a given, uniquely occurring, rule is shown in Equation (13) and is based on the calculation described in [83]. In our approach it is used to identify and eliminate duplicate instances by compressing the rule base into a set of M unique rules which are modelling the data.

$$scFuzzSup(\underline{FR_l}) = \frac{Co_{\underline{FR_l}}}{Co_{\underline{FR_l}} + Co_{\widehat{FR_l}}} \quad (13)$$

where l =1 to M , l is the index of the rule, $\underline{\boldsymbol{FR_l}}$ is a unique antecedent combination associated with the consequent linguistic label $\boldsymbol{B}$, $Co_{\underline{FR_l}}$ is the number of instances which support the rule $\underline{\boldsymbol{FR_l}}$ in the data set, $\widehat{\boldsymbol{FR_l}}$ is the set of contradictory antecedent combinations (the antecedent combinations which are different to $\underline{\boldsymbol{FR_l}}$ but have the same consequent as of $\underline{\boldsymbol{FR_l}}$), $Co_{\widehat{FR_l}}$ is the number of the instances which support these other combinations $\widehat{\boldsymbol{FR_l}}$.

The confidence of a rule is a measure of the validity of a rule which describes how tightly data patterns are associated to a specific output (Consequent). The confidence value is between 0 and 1. A confidence of 1 means that the pattern described in the rule is completely unique to a single output set. A confidence of less than 1 means that the pattern described in the rule occurs with more than one output set, and would then be associated with the output set with the highest confidence. The rule scaled confidence is calculated as shown in equation (13) and is based on the calculation in [48].

$$scConf(\underline{FR_l}) = \frac{scFuzzSup(\underline{FR_l})}{Co_{\widehat{RF_l}}} \quad (14)$$

Step3, *Calculation of Scaled Rule Weights*, in this step, the product of the scaled fuzzy support and confidence of a rule is used to calculate the scaled fuzzy weight of the rule as shown in Equation (15).

$$scWi = scFuzzSup \times scConf \quad (15)$$

Each of the generated $\boldsymbol{M}$ rules is assigned the scaled fuzzy weight measure $\boldsymbol{scWi}$ as follows:

$$B\left(W_{s_y t_{mz}}\right), B\left(W_{u_k t_{mz}}\right), B\left(W_{d_g t_{mz}}\right) \rightarrow B(R_h)[scWi] \quad (16)$$

The scaled fuzzy weight measures the quality of each rule in modelling the data. It can be used to rank the top rules associated to each output set and choose a single winner rule among compatible rules based on methods for rule weight specification [29]. We used these weights to extract the most representative rule patterns where the pattern with the highest value of $\boldsymbol{scWi}$ was selected over the other contradictive patterns. The selected patterns were used in a fuzzy system, as described in the following step, for modelling the relevancies based on the most important profile weighted terms.

Step 4, *Calculation of the Unified Term Weight*, in this step the resulting rules from the previous step are used to build a fuzzy system to calculate the unified term weight $\boldsymbol{W_{iykg}}$ for each query term $\boldsymbol{t_i}$ in each associated document visit $\boldsymbol{V_h}$. The fuzzy system calculates the unified term weight based on the term weights in the profiles of the associated user $\boldsymbol{U_k}$, document $\boldsymbol{D_g}$ and search task $\boldsymbol{S_y}$ which were created in phase 2 and the fuzzy rules extracted in step 3 of this phase.

The fuzzy system consists of the three input variables $\{\boldsymbol{W_{s_y t_{ez}}}, \boldsymbol{W_{u_k t_{ez}}}, \boldsymbol{W_{d_g t_{ez}}}\}$, one output variable which is the unified term weight $\boldsymbol{W_{iykg}}$, and the fuzzy rules which are extracted in step 3. The fuzzy system is then fed with the values of the inputs: $\boldsymbol{W_{s_y t_{ez}}}$, $\boldsymbol{W_{u_k t_{ez}}}$ and $\boldsymbol{W_{d_g t_{ez}}}$ which were associated with each query term for each document visit in step 1. The calculated value of $\boldsymbol{W_{iykg}}$ was then used to create the UTWI which consists of $\{\boldsymbol{V_h}, t_i, \boldsymbol{S_y}, \boldsymbol{U_k}, \boldsymbol{D_g}, \boldsymbol{W_{iykg}}\}$. UTWI was used in the next phase to create the recommendations.

Phase 5: Recommendation of Documents and Experts.

This phase creates a list of recommended documents and people (experts) based on a new user query and the UTWI. The new user query is pre-processed and transformed in a set of terms in the same way as in phase 1. Following that, the UTWI index is searched for the extracted query terms to find matching documents and experts who visited those documents frequently (i.e. the so called experts in that area). This starts with matching tasks to the user query. A matching task should have at least one occurrence of one of the query terms in its associated instances in UTWI. For each of these tasks the averages of the matching terms' weights are calculated. Then these weights are summed to give the aggregate task weight. Based on the aggregate task weight, the relevant documents and users of each matching task are extracted. A relevant document/user should have at least one matching term occurrence in UTWI with the task's terms. The average weight of each matching term is calculated for the relevant document/user and these are summed to calculate the aggregate weight of the relevant document/user. The document/users are sorted in descending order based on their aggregate weights.

Phase 6: Recommendation Presentation

In this phase, the recommended document and Experts are presented to the user through a web-based Graphical User Interface GUI). The GUI enables the user to view the recommended documents as a weighted list in which each document is associated with its relevance weight to the user query. It also shows a user analysis chart in which the relevant experts to the user query and their relevance weight are graphically represented. The GUI provides a query-task tree in which the relevant experts are grouped based on tasks which are relevant to the user query.

IV. EXPERIMENTS AND RESULTS

A. *Experimental Set up*

The standard test collection TREC Enterprise Track-2007 [75] 19] was used for the following experiments. The dataset consists of 370715 documents, with a total size 4.2 gigabytes. The corpus contains different types of documents such as html, text, pdf and others. The data set labelled as the test collection provides a group of 50 queries, which were created by real users and associated with the relevant documents for each query according to user judgement. The data set was extended by creating search tasks based on the labelled data and was indexed using a configured text based search system. The configured system is based on open source technology and consists of the following components: Apache Solr, Apache Tika and Hadoop. Hadoop is an open source framework for distributed computing. Tika is an open source toolkit that can parse and acquire different types of documents. Solr is an open source enterprise IS&R server, based on the underlying search library Lucene that is widely used in information retrieval applications. The system was developed to allow users to search for the provided tasks through a GUI. The system was configured to capture the relevance feedback (both implicit and explicit) from the users. In order to make the system easy to access for the participants, it was deployed on a web server and hosted by the Amazon Elastic Compute Cloud (EC2), which is part of Amazon Web Services (AWS) [5] [47][23]. *Fig. 8* shows the overarching architecture of the system.

3Thirty-five users were selected randomly and invited to participate in the experiments. Each user was briefed about the research aim and objectives, as well as the purpose of the user study. The experimental procedure was explained to users, including the steps within the experiment, the estimated time to complete the steps and how the captured data would be used in the research. The search tasks and the corresponding information sheet were given to the users to read. The tasks were then explained to the users. The users were then trained to use the system and asked to freely formulate their queries to search for information to help them find solutions to the given tasks.

All users were unpaid participants, consisting of 10 females and 25 males. The users had two occupations, university students and university staff members, and they were from various disciplines and qualified with various degrees (including undergraduate, postgraduate and doctorate). TABLE VII shows the characteristics of the users.

TABLE VI: PARTICIPANTS' CHARACTERISTICS.

Characteristic	***Category***	***#Of Participants***
Gender	Female	10
	Male	25
Age	(20-30)	24
	(30-40)	10
	(40-50)	1
Occupation	University Student	24
	University Staff Member	11
Education	School/ College	6
	Degree	8
	Masters	14
	PhD	7

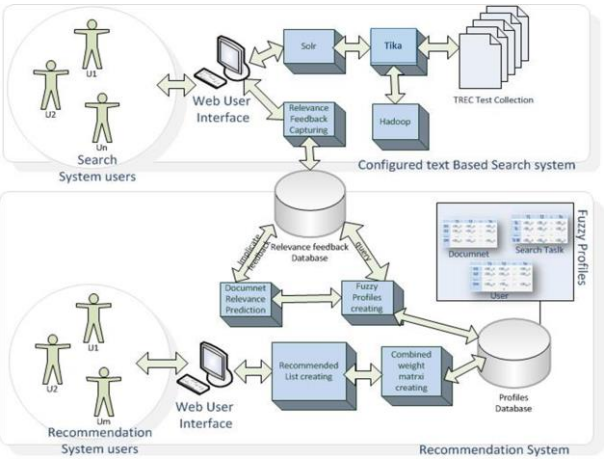


Fig. 8: System Architecture

As discussed in *Section III – Phase 1*, the system harvested 812 user queries and 1230 document visits, which gave a reasonable size of relevance feedback for creating the task, user, and document profiles. The captured implicit and explicit parameter values were used to develop the predictive linear model. Then, the captured user queries were pre-processed using the Oracle Text Search library. The extracted query terms were fed into the fuzzy system as discussed in *Section III– Phase 3*. The resulting profiles were projected and associated with the predicted document relevance levels. The resulting dataset instances were then mapped to linguistic labels in order to extract the fuzzy rules. The fuzzy rules were summarized as discussed in *Section III – Phase 4*. The summarized fuzzy rules which are shown in TABLE VIII were used to build a fuzzy inference system for calculating the combined term weights.

.

TABLE VII: SAMPLE OF THE SUMMARIZED WEIGHTED FUZZY RULES

W_T	W_U	W_D		W_R	$scWi$
H	H	H	→	H	0.489292903
H	H	M	→	H	0.364747958
H	L	H	→	M	0.116836792
H	L	M	→	M	0.082494544
H	M	H	→	H	0.302517053
H	M	L	→	M	0.010468469
	…	…	…	..	…..

A web based GUI was developed to handle the user queries as discussed in *Section. III Phase 6*.

V. EVALUATION

The proposed system was evaluated using three well-known methods:

- Linear Predictive Model Validation
- Hold-Out method
- Comparative Retrieval Performance Analysis

A. *Linear Predictive Model Validation*

The model accuracy was validated using the R-squared ($\boldsymbol{R^2}$) measurement. $\boldsymbol{R^2}$ measure is one of the most widely used and reported measures of the accuracy of the statistical models [27]. $\boldsymbol{R^2}$ is formally calculated as shown in Equation (17).

$$R^2 = \frac{\sum_{i=1}^{n}(\hat{y} - \bar{y})^2}{\sum_{i=1}^{n}(y - \bar{y})^2} = \frac{SS\ Predicted}{SS\ total} \qquad (17)$$

where $\boldsymbol{n}$ is the number of the observation (data instances), $\hat{y}$ is the predicted value of the data instance i , $\bar{y}$ is the mean of the actual values of data instances in the set, y is the actual value of the data instance i, and SS is the sum of the squares.
As shown in TABLE IX, the accuracy of the predictive linear model was 76.5 %. However, the accuracy can be improved through the adaptive mechanism of the proposed model when more relevance feedback is captured.

TABLE IX: SUM SQUARES FOR THE LINEAR MODEL

Source	S- Squares	df	M- Square	f
Predicted	9,510.566	5	1,902.173	747.94
Residuals	2,901.812	1,141	2.543	
Total	12,412.678	1,146		
Accuracy	76.5 %			

Fig. 9, visualizes the linear relation between the estimated or predicted values and the observed or actual values of the dependent variable (explicit relevance level). The figure shows clearly that the pivot points between the predicted and actual values are almost distributed in a linear form, which reflects the accuracy of the predictive model.

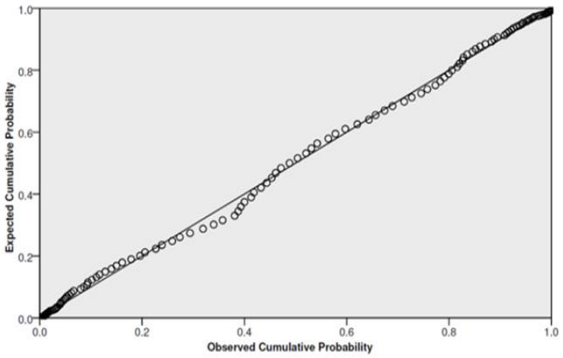


Fig. 9: Pivot of the Predicted Value and Actual Value

B. Hold-Out method

The developed fuzzy rule based system was validated using the Hold-Out method [7]. In this method a dataset D is divided into two subsets, $\boldsymbol{D_t}$ (usually 80% of $\boldsymbol{D}$) and $\boldsymbol{D_h}$ (usually 20% of $\boldsymbol{D}$). Set $\boldsymbol{D_t}$, named as the training set, is used to train the model while D_h, called the testing set or hold-out set is used to test the model in order to calculate the accuracy.

Formally let $\boldsymbol{D_h}$ be a subset of $\boldsymbol{D}$ of size $\boldsymbol{h}$ and let $\boldsymbol{D_t}$ be D - $\boldsymbol{D_h}$, then the Hold-out estimated accuracy is defined as:

$$\mathrm{acc_h} = \frac{1}{\mathrm{h}} \sum_{(\mathrm{v_i},\mathrm{y_i})\in \mathrm{D_h}} \sigma(\mathrm{v_i}, \mathrm{y_i}) \tag{18}$$

where $\boldsymbol{\sigma(v,y) = 1}$ if $v=y$ and 0 otherwise, $\boldsymbol{v_i}$ is the predicted value of the instance $\boldsymbol{i}$, $\boldsymbol{y_i}$ is the actual value of the instance $\boldsymbol{i}$.

The rule extraction and summarizing components were trained on 80% of the data set and then it was tested on the unseen 20% of the data set. The resulting rules were used to classify the relevancy of each instance in the unseen data as described in step 4 of *Section III– Phase 4*. The resulting relevancy classifications were compared with the associated linguistic labels of the actual explicit relevance feedback values as shown in TABLE X. These linguistic labels used the same fuzzy sets as in step 1 of *Section III– Phase 4*. The system shows 86% performance accuracy in correctly classifying documents relevance. This produces good initial performance, which can further be improved through the adaptive mechanism of the proposed system.

TABLE VII: SAMPLE OF SUMMARISED FUZZY RULES ACCURACY

Term	Task	User	Document	Predicted Relevance	Actual Relevance	Match
Cotton	T13	**@coventry.ac.uk	04574741.html	M	M	1
Tech	T1	**@coventry.ac.uk	04587909.html	M	L	0
Air	T12	**@uni.coventry.ac.	04736857.html	M	M	1
Guitar	T13	**@yahoo.com	12228999.html	M	M	1
Australia	T12	**@coventry.ac.uk	02493670.html	M	M	1
Australia	T13	**@gmail.com	03007618.html	M	M	1
School	T1	**@coventry.ac.uk	16400222.html	M	L	0
Cooper	T13	**@coventry.ac.uk	16400222.html	M	M	1
Cooper	T13	**@yahoo.com	16400222.html	M	M	1
Bio	T12	**@yahoo.com	16456196.html	M	M	1
Lab	T13	**@gmail.com	08225989.html	M	M	1

C. Comparative performance retrieval evaluation

In order to evaluate the overall retrieval performance of the proposed system, a comparative evaluation was conducted to compare the retrieval performance measures of the proposed system with the existing standard Solr search system and the semantic based enterprise IS&R Lucid [58]. Both systems were based on Solr as a core search platform; however, the first one used the standard inverted index while the second used semantic indexing [58]. The standard inverted index consists of the terms associated with their frequencies in the indexed documents. In semantic indexing the terms are given semantic weights to reflect their relevance to the indexed documents. Precision (***P***) and Recall (***R***), which are standard evaluation metrics used in information retrieval research evaluation [25] and document

ranking analysis, were used to measure the retrieval performance of the three systems. Precision (***P***) and Recall (***R***) are used to test the ability of the system to retrieve the relevant documents while the document ranking analysis was used to test the ability of the system to give a high ranking to the relevant document in the search result.

C.1. ***Precision and Recall Analysis***

Precision (***P***) is a measure of the ability of a system to present only relevant items. P is defined as follows:

$$P = \frac{|Ra|}{|A|} \tag{19}$$

where ***Ra*** is the number of relevant items retrieved and ***A*** is the total number of items retrieved in response to a user query.

Recall (***R***) is a measure of the ability of a system to present all relevant items. ***R*** is defined as follows:

$$R = \frac{|Ra|}{|Rm|} \tag{20}$$

where ***Ra*** is the number of relevant items retrieved and ***Rm*** is the total number of relevant items in the document set.

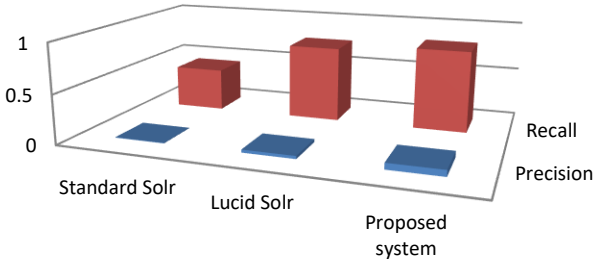
The evaluation was carried out using the first 25 of the provided queries in the labelled data as these were used for creating the simulated search tasks in the user study. The results are shown in TABLE XI. The queries were given to the three search systems, Standard Solr, Lucid and the proposed system. P and R were calculated for each of the given queries for each system. Then the averages P and R were calculated for each system.

TABLE VIIII: PRECISION (P) AND RECALL (R) FOR: STANDARD VECTOR SPACE SEARCH SYSTEM (STANDARD SOLR), SEMANTIC BASED SEARCH SYSTEM (LUICID SOLR) AND THE PROPOSED SYSTEM.

	Precision(P)			***Recall (R)***		
QUERY ID	***Sandard Solr***	***Lucid Solr***	***Proposed System***	***Standard Solr***	***Lucid Solr***	***Proposed System***
CE-001	0	0.006	0.023	0	0.5	0.667
CE-002	0.029	0.004	0.036	0.667	1	0.667
CE-003	0	0.005	0.061	0	1	1
CE-004	0	0.063	0.063	0	0.333	0.667
CE-005	0	0.007	0.035	0	0.5	0.667
CE-006	0	0.006	0.097	0	0.75	0.75
CE-007	0.006	0.105	0.105	0.2	0.727	0.727
CE-008	0.004	0.023	0.188	0.308	0.692	0.692
CE-009	0.003	0.163	0.163	0.1	0.7	0.7
CE-010	0.007	0.041	0.041	0.5	1	1
CE-011	0	0.008	0.008	0	1	1
CE-012	0.001	0.057	0.057	0.333	0.5	0.5
CE-013	0	0.02	0.02	0	0.333	0.333
CE-014	0.004	0.023	0.023	1	0.333	1
CE-015	0.004	0.012	0.019	1	1	1
CE-016	0.022	0.002	0.036	1	1	1
CE-017	0	0.003	0.015	0	1	1
CE-018	0	0.007	0.038	1	1	1
CE-019	0	0.009	0.049	0	0.667	0.5
CE-020	0.004	0.003	0.036	1	1	1
CE-021	0.003	0.008	0.12	0.667	1	1
CE-022	0.014	0.031	0.075	0.667	0.333	1
CE-023	0.001	0.033	0.094	1	1	1
CE-024	0.005	0.035	0.063	0.8	1	1
CE-025	0	0.071	0.132	0.667	0.833	0.833
AVG	0.00428	0.0298	0.06388	0.43636	0.76804	0.82812

As shown in *Fig. 10*, the average ***P*** value for a standard Solr system was 0.00428, which is relatively low. This low value of ***P*** indicated that the system retrieved a large number of irrelevant documents. The ***P*** value for the semantic based search Lucid was 0.0298, which was significantly higher but still indicated a large number of retrieved documents. The proposed approach improved the retrieval accuracy, shown through the increase inaverage value of ***P*** to 0.064. In other words, the proposed approach reduced the number of irrelevant documents in the search result, which meant that the ability of the system to show only the relevant documents was enhanced. Referring to Equation (19), the increment on ***P*** can either be due to the increase in the number of relevant documents retrieved or because of a decrease in the number of non-relevant documents retrieved or the combination of both of these factors. The proposed approach increased the number of relevant items retrieved and also lowered the number of non- relevant items retrieved because the recommended

or retrieved document list was filtered more effectively based on the fuzzy rule based profiling.



	Standard Solr	Lucid Solr	Proposed system
Precision	0.00428	0.0298	0.06388
Recall	0.43636	0.76804	0.82812

Fig.10: Precision (P) and Recall (R) for: Standard Inverted Index System (Standard Solr); Semantic Based Search System (Luicid Solr); and the Proposed Recommender System

Fig. 10 shows that the proposed system outperformed the other two systems as it achieved a significantly higher average value of R. This indicates the enhanced ability of the proposed system to retrieve the relevant documents based on the user query and the semantic relationship among the tasks, users and documents.

C.2. ***Document Ranking Analysis***

The search system was not only required to retrieve the relevant documents, but also to show it at the top of the search engine result page (SERP) [2][22], and the search engines try to show the relevant information in the first or proceeding ten results [29]. Precision and recall ratios do not indicate the ranking of the document or whether or not the document was shown in the first 10 documents. Therefore, a comparative document ranking analysis was carried out to find out if the proposed approach has improved the ability of the system to give a high ranking to the retrieved relevant documents and to push them to the top of the SERP. In this analysis, the ranking of the relevant documents for the first 25 queries, which were used to create the search tasks for data capturing, was divided into 7 categories which were: A (1-10), B (11-20), C (21 -30), D (31-40), E (>40), F (Not retrieved). Following this the number of relevant documents which fell into each category was calculated. These frequencies were then compared to the frequencies resulting from the standard Solr and Lucid search systems for the same set of queries as shown in Table XII.

TABLE IXI: COMPARED DOCUMENT FREQUENCIES FOR RANK CATEGORIES

	Number of relevant documents			**Percentage of relevant document**		
Ranking Category	***StdSolr***	***Lucid Sor***	***Proposed***	***StdSolr***	***Lucid Sor***	***Proposed***
A (1 – 10)	15	50	60	15.96%	53.19%	63.83%
B (11 – 20)	13	13	14	13.83%	13.83%	14.89%
C (21 – 30)	4	7	2	4.26%	7.45%	2.13%
D (31- 40)	7	2	1	7.45%	2.13%	1.06%
E (> 41)	14	1	1	14.89%	1.06%	1.06%
F (Not Retrieved)	41	21	16	43.62%	22.34%	17.02%
Total	***94***	***94***	***94***	***100%***	***100%***	***100%***

As shown in *Fig. 11*, the proposed approach has improved the system's ranking performance. As compared to the standard Solr, the percentage of documents ranked in category A has increased from 15.96% to 63.83%. Also, the percentage of category D and E, which are not relevant, has dropped down significantly from 7.45% to 1.06% and from 14.89% to 1.06%. Finally, the category F percentage has decreased from 43.62% to 17.02%. The proposed system has achieved a better ranking performance than the semantic based search Lucid where the system has showed more relevant documents in category A and B than Lucid.

The improvement in the relevant document ranking is due to the user relevance feedback that was used to train and validate the system which together with the help of the user judgement pushed the relevant documents to the top of SERP. *Fig. 12* shows a snapshot of the developed system.

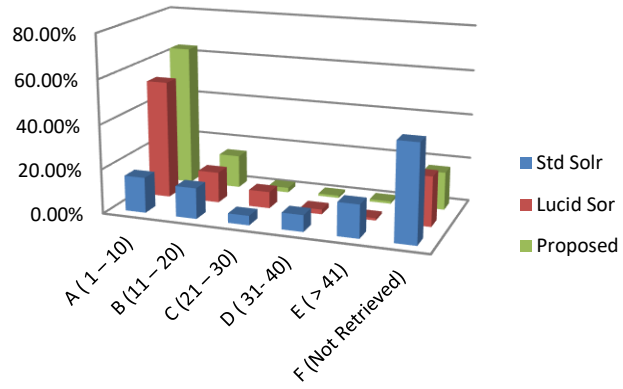


Fig. 11: Compared Document Frequencies for Rank Categories

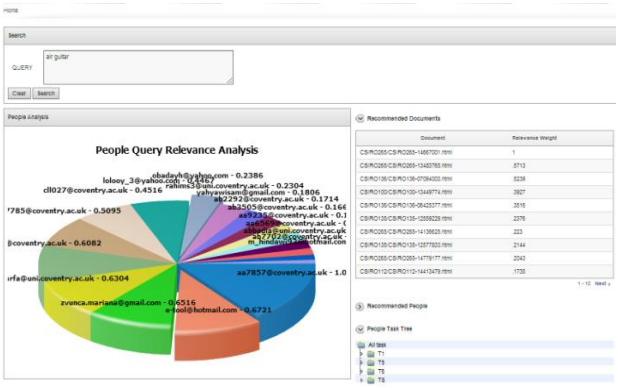


Fig. 12: Snapshot of the Developed System

VI. Conclusion and Future Work

In this paper, a new approach for the development of a fuzzy logic based profiling approach for an enterprise system using enterprise IS&R was presented. The approach provides a new mechanism for developing and integrating task, user and document profiles into a unified index through the use of relevance feedback and a fuzzy rule based summarisation technique. The motivation for using the fuzzy approach was to handle the uncertainty in the data caused due to inconsistency and subjectivity in the assessment of relevance by the user.

Experiments in which the relevance feedback was captured from 35 users on 20 predefined simulated enterprise IS&R tasks were successfully run. During this process the system captured implicit and explicit feedback parameters, and the user queries. The captured dataset was used to develop and train the fuzzy system. The empirical research carried out as part of this research clearly found significant co-relation between the implicit and explicit relevance feedback. It was found that there was a linear relationship between the implicit parameters (i.e. time on page, mouse movements, and mouse clicks) and the explicit document relevance. The linear relationship was then translated into an adaptive linear predictive model to estimate the document relevance from the implicit feedback parameters with an accuracy performance of 76%. Theexperiments showed that incorporating user query terms together with the implicit parameters in the proposed fuzzy based mechanism improved the predictive accuracy. The system showed 86% accuracy in correctly classifying document relevance. This performance can be further improved through the adaptive mechanism of the proposed system by involving more users and search tasks. Considering the search context by building three profiles and then combining them in one index through a fuzzy based mechanism also contributed to enhance the predictive accuracy of the proposed system.

The overall performance of the proposed approach was evaluated based on standard precision and recall metrics which showed significant improvements in retrieving relevant documents. A comparative evaluation was carried out with two existing systems, standard Solr and Lucid. In this evaluation, the proposed system outperformed both in the manner of the retrieval accuracy as shown in the results section.

Our future work will include further evaluation of the proposed approach in a real world organisation using an extended user base over a longer time scale to improve the performance of the system. We also plan to investigate more parameters related to user information behaviour in a real world enterprise situation.

VII. REFERENCES

[1]. U. Afzal, M. Islam, Meven: An Enterprise Trust Recommender System (2013).

[2]. E. Agichtein, E. Brill, S. Dumais, R. Ragno, Learning user interaction models for predicting web search result preferences, In: Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval Anonymous , ACM (2006) 3-10.

[3]. S. Akuma, R.Iqbal, C.Jayne, F.Doctor, "Comparative analysis of relevance feedback methods based on two user studies", Computers in Human Behaviour, Elsevier 60 (2016) 138-146.

[4]. M.Y.H. Al-Shamri, N.H. Al-Ashwal, Fuzzy-weighted similarity measures for memory-based collaborative recommender systems, Journal of Intelligent Learning Systems and Applications (2014).

[5]. Amazon, Elastic Compute Cloud (EC2) Cloud Server & Hosting – AWS (2015).

[6]. D. Anand, B.S. Mampilli, Folksonomy-based fuzzy user profiling for improved recommendations, Expert Systems with Applications 41 (2014) 2424-2436.

[7]. S. Arlot, A. Celisse, A survey of cross-validation procedures for model selection, Statistics surveys 4 (2010) 40-79.

[8]. Z. Attia, A. Gadallah, and H. Hefny. "An enhanced multi-view fuzzy information retrieval model based on linguistics." IERI Procedia 7 (2014) 90-95.

[9]. R. Baeza, B. Ribeiro-Neto, Enterprise Search, In: Modern Information Retrieval 2nd Ed Anonymous, Pearson Educational (2011) 641-44.

[10]. V.Balakrishnan, X. Zhang, Implicit user behaviours to improve post-retrieval document relevancy, Computers in Human Behavior 33 (2014) 104-112.

[11]. K. Balog, L. Azzopardi, M. De Rijke, Formal models for expert finding in enterprise corpora, In: Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval Anonymous , ACM (2006) 43-50.

[12]. K. Balog, L. Azzopardi, M. de Rijke, A language modelling framework for expert finding, Information Processing & Management 45 (2009) 1-19.

[13]. Z. Bao, B. Kimelfeld, Y. Li, Automatic suggestion of query-rewrite rules for enterprise search, In: Proceedings of the 35th international ACM SIGIR conference on Research and development in information retrieval Anonymous, ACM (2012) 591-600.

[14]. P. Bidoki, A.M.Z. Ghodsnia, N. Yazdani, F. Oroumchian, A3CRank: An adaptive ranking method based on connectivity, content and click-through data, Information processing & management 46 (2010) 159-169. [3]

[15]. A.Z. Broder, A.C. Ciccolo, Towards the next generation of enterprise search technology, IBM Systems Journal 43 (2004) 451-454.

[16]. G. Buscher, R.W. White, S. Dumais, J. Huang, Large-scale analysis of individual and task differences in search result page examination strategies, In: Proceedings of the fifth ACM international conference on Web search and data mining Anonymous , ACM (2012) 373-382.

[17]. F.J. Cabrerizo , R. Ureña, W. Pedrycz, E. Herrera-Viedma, Building consensus in group decision making with an allocation of information granularity. Fuzzy Sets and Systems 255 (2014) 115-127.

[18]. L.M. Campos, J.M. Fernández-Luna, J.F. Huete, A collaborative recommender system based on probabilistic inference from fuzzy observations, Fuzzy Sets and Systems 159 (2008) 1554-1576.

[19]. Y. Cao, J. Liu, S. Bao, H. Li, Research on Expert Search at Enterprise Track of TREC 2005, In: TREC Anonymous (2005).

[20]. J. Carbo, J.M. Molina, Agent-based collaborative filtering based on fuzzy recommendations, International journal of Web engineering and technology 1 (2004) 414-426.

[21]. G. Castellano, A.M. Fanelli, C. Mencar, M.A. Torsello, Similarity-based fuzzy clustering for user profiling, In: Proceedings of the 2007 IEEE/WIC/ACM International Conferences on Web Intelligence and Intelligent Agent Technology-Workshops Anonymous , IEEE Computer Society (2007) 75-78.

[22]. M. Castellanos, Hotminer: Discovering hot topics from dirty text, In: Survey of Text Mining Anonymous , Springer (2004) 123-157.

[23]. S. Chaisiri, R. Kaewpuang, B. Lee, D. Niyato, Cost minimization for provisioning virtual servers in amazon elastic compute cloud, In: Modeling, Analysis & Simulation of Computer and Telecommunication Systems (MASCOTS), 2011 IEEE 19th International Symposium on Anonymous , IEEE (2011) 85-95.

[24]. M. Claypool, D. Brown, P. Le, M. Waseda, Inferring user interest, Internet Computing, IEEE 5 (2001) 32-39.

[25]. K. Collins-Thompson, P.N. Bennett, R.W. White, S. de la Chica, D. Sontag, Personalizing web search results by reading level, In: Proceedings of the 20th ACM international conference on Information and knowledge management Anonymous , ACM (2011) 403-412.

[26]. C. Cornelis, J. Lu, X. Guo, G. Zhang, One-and-only item recommendation with fuzzy logic techniques, Information Sciences 177 (2007) 4906-4921.
[27]. C. Dan, C. Sherlock, Introduction to Regression and Data Analysis, October 28, (2008). [9]
[28]. [32]Delic, K.A. J.A. Riley, Enterprise Knowledge Clouds: Next Generation KM Systems?., In: eKNOW Anonymous (2009) 49-53.
[29]. H. Deng, I. King, M.R. Lyu, Formal models for expert finding on dblp bibliography data, In: Data Mining, 2008. ICDM'08. Eighth IEEE International Conference on Anonymous, IEEE (2008) 163-172.
[30]. H. Deng, J. Han, M.R. Lyu, I. King, Modeling and exploiting heterogeneous bibliographic networks for expertise ranking, In: Proceedings of the 12th ACM/IEEE-CS joint conference on Digital Libraries Anonymous, ACM (2012) 71-80.
[31]. P.A. Dmitriev, N. Eiron, M. Fontoura, E. Shekita, Using annotations in enterprise search, In: Proceedings of the 15th international conference on World Wide Web Anonymous, ACM (2006) 811-817.
[32]. F. Doctor, H. Hagras, D. Roberts, V. Callaghan, A fuzzy based agent for group decision support of applicants ranking within recruitment systems, In: Intelligent Agents, 2009. IA'09. IEEE Symposium on Anonymous, IEEE (2009) 8-15.
[33]. A. Eckhardt, Similarity of users'(content-based) preference models for Collaborative filtering in few ratings scenario, Expert Systems with Applications 39 (2012) 11511-11516.
[34]. R. Fagin, R. Kumar, K.S. McCurley, J. Novak, D. Sivakumar, J.A. Tomlin, D.P. Williamson, Searching the workplace web, In: Proceedings of the 12th international conference on World Wide Web Anonymous , ACM (2003), pp. 366-375.
[35]. S. Feldman, The high cost of not finding information, Information Today, Incorporated (2004).
[36]. S.D. Gollapalli, P. Mitra, C.L. Giles, Similar researcher search in academic environments, In: Proceedings of the 12th ACM/IEEE-CS joint conference on Digital Libraries Anonymous, ACM (2012) 167-170.
[37]. A. Grzywaczewski, Iqbal R., Task-specific information retrieval systems for software engineers, Journal of Computer and System Sciences 78 (2012) 1204-1218.
[38]. Q. Guo, E. Agichtein, Beyond dwell time: estimating document relevance from cursor movements and other post-click searcher behavior, In: Proceedings of the 21st international conference on World Wide Web Anonymous, ACM (2012) 569-578.
[39]. Y. Gupta, A. Saini, A. Saxena, A new fuzzy logic based ranking function for efficient information retrieval system, Expert Systems with Applications 42 (2015) 1223-1234.
[40]. Y. Gupta, A. Saini, A. Saxena, Fuzzy logic-based approach to develop hybrid similarity measure for efficient information retrieval, Journal of Information Science (2014).
[41]. K.A. Gust, M.L. Mayo, Z.A. Collier, Ranking the Relative Importance of Toxicological Observations Based on Subject Matter Expertise (2015).
[42]. D. Hawking, Challenges in enterprise search, In: Proceedings of the 15th Australasian database conference-Volume 27 Anonymous, Australian Computer Society, Inc., 2004, pp. 15-24.
[43]. IBM, IBM Knowledge Center – Importance (2015).
[44]. IBM, IBM Knowledge Center- Regression Model (2015).
[45]. R. Iqbal, A. Grzywaczewski, J. Halloran, F. Doctor, K. Iqbal, Design implications for task-specific search utilities for retrieval and re-engineering of code, Enterprise Information Systems (2015) 1-20.
[46]. R. Iqbal, F. Doctor, N. Shah, X. Fei., An intelligent framework for activity led learning in network planning and management, Journal of Computer Networks and Distributed Systems, 12 (2014) 401-419.
[47]. R. Iqbal, , N. Shah, F. Doctor, A. James, , T. Cichowicz, , Integration, optimization and usability of enterprise applications, Proceedings of the 16th International Conference on CSCW in Design, IEEE Press, (2012) 431-437.
[48]. H. Ishibuchi, T. Yamamoto, Rule weight specification in fuzzy rule-based classification systems, Fuzzy Systems, IEEE Transactions on 13 (2005) 428-435.
[49]. D. Jannach, L. Lerche, M. Jugovac, Adaptation and Evaluation of Recommendations for Short-term Shopping Goals, In: Proceedings of the 9th ACM Conference on Recommender Systems Anonymous , ACM, 2015, pp. 211-218.
[50]. G. Jawaheer, M. Szomszor, P. Kostkova, Comparison of implicit and explicit feedback from an online music recommendation service, In: proceedings of the 1st international workshop on information heterogeneity and fusion in recommender systems Anonymous , ACM (2010) 47-51.
[51]. S. Jung, J.L. Herlocker, J. Webster, Click data as implicit relevance feedback in web search, Information Processing & Management 43 (2007) 791-807.
[52]. D. Kelly, Methods for evaluating interactive information retrieval systems with users, Foundations and Trends in Information Retrieval 3 (2009) 1-224.
[53]. J. Kim, D.W. Oard, K. Romanik, Using implicit feedback for user modeling in internet and intranet searching (2000).
[54]. L. Li, L. Zheng, F. Yang, T. Li, Modeling and broadening temporal user interest in personalized news recommendation, Expert Systems with Applications 41 (2014) 3168-3177.

[55]. Q. Li, B.M. Kim, Constructing user profiles for collaborative recommender system, In: Advanced Web Technologies and Applications Anonymous, Springer (2004) 100-110.
[56]. X. Liu, H. Fang, F. Chen, M. Wang, Entity centric query expansion for enterprise search, In: Proceedings of the 21st ACM international conference on Information and knowledge management Anonymous, ACM (2012) 1955-1959.
[57]. Y. Liu, J. Miao, M. Zhang, S. Ma, L. Ru, How do users describe their information need: Query recommendation based on snippet click model, Expert Systems with Applications 38 (2011) 13847-13856.
[58]. LucidWorks, lucidworks (2015).
[59]. C. Macdonald, I. Ounis, Voting techniques for expert search, Knowledge and information systems 16 (2008) 259-280.
[60]. C. Martinez-Cruz, C. Porcel, J. Bernabé-Moreno, E. Herrera-Viedma, A model to represent users trust in recommender systems using ontologies and fuzzy linguistic modeling. Information Sciences 311 (2015) 102-118. [
[61]. M. Morita, Y. Shinoda, Information filtering based on user behavior analysis and best match text retrieval, In: Proceedings of the 17th annual international ACM SIGIR conference on Research and development in information retrieval Anonymous , Springer-Verlag New York, Inc. (1994) 272-281.
[62]. R. Mukherjee, J. Mao, Enterprise search: Tough stuff, Queue 2 (2004) 36.
[63]. L. Page, S. Brin, R. Motwani, T. Winograd, The PageRank citation ranking: Bringing order to the web. (1999).
[64]. W. Pedrycz, Granular Computing: Analysis and Design of Intelligent Systems, CRC Press/Francis Taylor, Boca Raton (2013).
[65]. P. Perny, J. Zucker, Preference-based search and machine learning for collaborative filtering: the'Film-Conseil'movie recommender system, Revue I3 1 (2001) 1-40.
[66]. B. Poblete, R. Baeza-Yates, Query-sets: using implicit feedback and query patterns to organize web documents, In: Proceedings of the 17th international conference on World Wide Web Anonymous, ACM (2008) 41-50.
[67]. M.F. Porter, An algorithm for suffix stripping, Program 14 (1980) 130-137.
[68]. P.Raghavan, Structured and unstructured search in enterprises, IEEE Data Eng.Bull. 24 (2001) 15-18.
[69]. P. Roy, A novel fuzzy document-based information retrieval scheme (FDIRS). In Applied Informatics 3 (2016) 1.
[70]. A. Saini, Y. Gupta, A.K. Saxena, Fuzzy Based Approach to Develop Hybrid Ranking Function for Efficient Information Retrieval. In Advances in Intelligent Informatics, Springer (2015) 471-479.
[71]. S. Schiaffino, A. Amandi, Intelligent user profiling, In: Artificial Intelligence An International Perspective Anonymous, Springer (2009) 193-216.
[72]. M. Seleng, M. Laclavik, S. Dlugolinsky, M. Ciglan, M. Tomasek, L. Hluchy, Approach for enterprise search and interoperability using lightweight semantic, In: Intelligent Engineering Systems (INES), 2014 18th International Conference on Anonymous , IEEE (2014) 73-78.
[73]. P. Singh, S. Dhawan, S. Agarwal, & N. Thakur, Implementation of an efficient Fuzzy Logic based Information Retrieval System. arXiv preprint arXiv:1503.03957 (2015).
[74]. M. Sun, F. Li, J. Lee, K. Zhou, G. Lebanon, H. Zha, Learning multiple-question decision trees for cold-start recommendation, In: Proceedings of the sixth ACM international conference on Web search and data mining Anonymous , ACM (2013) 445-454.
[75]. Text Retrieval Conference, Text Retrieval Conference (TREC) 2007 Enterprise Track (2015). http://trec.nist.gov/data/t16_enterprise.html
[76]. S.K. Tyler, J. Teevan, Large scale query log analysis of re-finding, In: Proceedings of the third ACM international conference on Web search and data mining Anonymous , ACM (2010) 191-200. [72]
[77]. S.K. Tyler, J. Wang, Y. Zhang, Utilizing re-finding for personalized information retrieval, In: Proceedings of the 19th ACM international conference on Information and knowledge management Anonymous , ACM (2010) 1469-1472.
[78]. P.C. Vaz, D. Martins de Matos, B. Martins, Stylometric relevance-feedback towards a hybrid book recommendation algorithm, In: Proceedings of the fifth ACM workshop on Research advances in large digital book repositories and complementary media Anonymous , ACM (2012) 13-16.
[79]. G.A. Wang , J. Jiao, A.S. Abrahams, W. Fan, Z. Zhang, ExpertRank: A topic-aware expert finding algorithm for online knowledge communities, Decision Support Systems 54 (2013) 1442-1451
[80]. W. Wang, L. Chen, A Class Similarity Based Weight Estimation Algorithm for the Personalized Enterprise Search Engines, Journal of Computational Information Systems 10 (2014) 1903-1910.
[81]. M. White, S.G. Nikolov, S. Monteleone, R. Compaño, I. Maghiros, Enterprise Search in the European Union: A Techno-economic Analysis, Publications Office (2013).
[82]. R.W. White, G. Buscher, Text selections as implicit relevance feedback, In: Proceedings of the 35th international ACM SIGIR conference on Research and development in information retrieval Anonymous, ACM (2012) 1151-1152.
[83]. D. Wu, J.M. Mendel, J. Joo, Linguistic summarization using IF-THEN rules, In: Fuzzy Systems (FUZZ), 2010 IEEE International Conference on Anonymous, IEEE (2010) 1-8.
[84]. R.R. Yager, Fuzzy logic methods in recommender systems, Fuzzy Sets and Systems 136 (2003) 133-149.

[85]. .D. Zhou S. Orshanskiy, H. Zha, C.L. Giles, Co-ranking authors and documents in a heterogeneous network, In: Data Mining. ICDM. Seventh IEEE International Conference on Anonymous , IEEE (2007) 739-744.